IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Roman Baranov
Sep 14, 2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- We have started this project with collecting data. Our two sources were SpaceX API and scraping Wikipedia
- We then examined, cleaned and wrangled obtained data with the help of Pandas and Numpy libraries
- We then moved on to exploring the data (EDA) with SQL queries and visualizations using the Seaborn and MatPlotLib libraries to build graphs
- Next we moved on to interactive visualizations by plotting locations on an interactive map using Folium library and building a dashboard using Dash and Plotly
- Finally after engineering useful features we moved on to modeling the data using different classifications algorithms and comparing their effectiveness

Introduction

The commercial space age is here, companies are making space travel affordable for everyone. Virgin Galactic is providing suborbital spaceflights. Rocket Lab is a small satellite provider. Blue Origin manufactures sub-orbital and orbital reusable rockets. Perhaps the most successful is SpaceX. SpaceX's accomplishments include: Sending spacecraft to the International Space Station. Starlink, a satellite internet constellation providing satellite Internet access. Sending manned missions to Space.

One reason SpaceX can do this is the rocket launches are relatively inexpensive.

We will predict if the Falcon 9 first stage will land successfully. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.

Therefore if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX API
  - Web scrape Wikipedia
- Perform data wrangling
  - Examined and cleaned initial data
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Built, tuned and evaluated SVM, Classification Trees, KNN and Logistic Regression classification models

Data Collection

Making HTTP requests(Requests library) to the SpaceX API allowed us to obtain:

- booster name
- the name of the launch site being used
- the longitude, and the latitude
- the mass of the payload and the orbit that it is going to
- the outcome of the landing, the type of the landing
- number of flights with that core
- whether gridfins were used
- whether the core is reused
- whether legs were used
- the landing pad used
- the block of the core which is a number used to separate version of cores
- the number of times this specific core has been reused
- and the serial of the core

We also used Beautiful Soup library to webscrape information about SpaceX launches from Wikipedia:

Data Collection – SpaceX API

GitHub URL of the completed SpaceX API calls:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W1L1_jupyter-labs-spacex-data-collection-api.ipynb

Import Libraries and Define Auxiliary Functions

↓

Request and parse the SpaceX launch data using the GET request

↓

Filter the dataframe to only include Falcon 9 launches

↓

Deal with Missing Values

Data Collection - Scraping

GitHub URL of the completed web scraping notebook:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W1L2_jupyter-labs-webscraping.ipynb

Request the Falcon9 Launch Wiki page from its URL

↓

Extract all column/variable names from the HTML table header

↓

Create a data frame by parsing the launch HTML tables

Data Wrangling

Performed exploratory Data Analysis and determined Training Labels

GitHub URL of completed data wrangling notebook:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W1L3_IBM-DS0321EN-SkillsNetwork_labs_module_1_L3_labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

Examine data types and missing data

↓

Calculate the number of launches on each site

↓

Calculate the number and occurrence of each orbit

↓

Calculate the number and occurence of mission outcome per orbit type

↓

Create a landing outcome label from Outcome column

EDA with Data Visualization

Scatter plots to visualize:

- FlightNumber vs. PayloadMassand
- FlightNumber vs LaunchSite
- Payload Vs. Launch Site
- FlightNumber and Orbit type
- Payload and Orbit type

(best way to display a mix of categorical and continuous variables)

Bar chart to visualize:

- success rate of each orbit type (easy to see which bar is taller)

Line chart to visualize:

- launch success yearly trend (a continuous line makes sense when plotting time data)

GitHub URL of completed EDA with data visualization notebook:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W2L2_IBM-DS0321EN-SkillsNetwork_labs_module_2_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first succesful landing outcome in ground pad was acheived
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

GitHub URL of completed EDA with SQL notebook:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W2L1_jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

To find geographical patterns about launch sites we have added each site's location on a map using site's latitude and longitude coordinates as a **Circle** and a **Marker**

Enhanced the map by adding the launch outcomes for each site in different colors and saw which sites have high success rates. Since many outcomes have the same coordinates we used a **MarkerCluster** object

To help with the exploration and analysis of the proximities of launch sites we added a **MousePosition** on the map to get coordinate for a mouse over a point on the map and drew a **PolyLine** between Launch sites and railway, highway, coastline, etc.

GitHub URL of completed interactive map with Folium:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W3_L1_IBM-DS0321EN-SkillsNetwork_labs_module_3_lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

To obtain visual insights about the following questions:

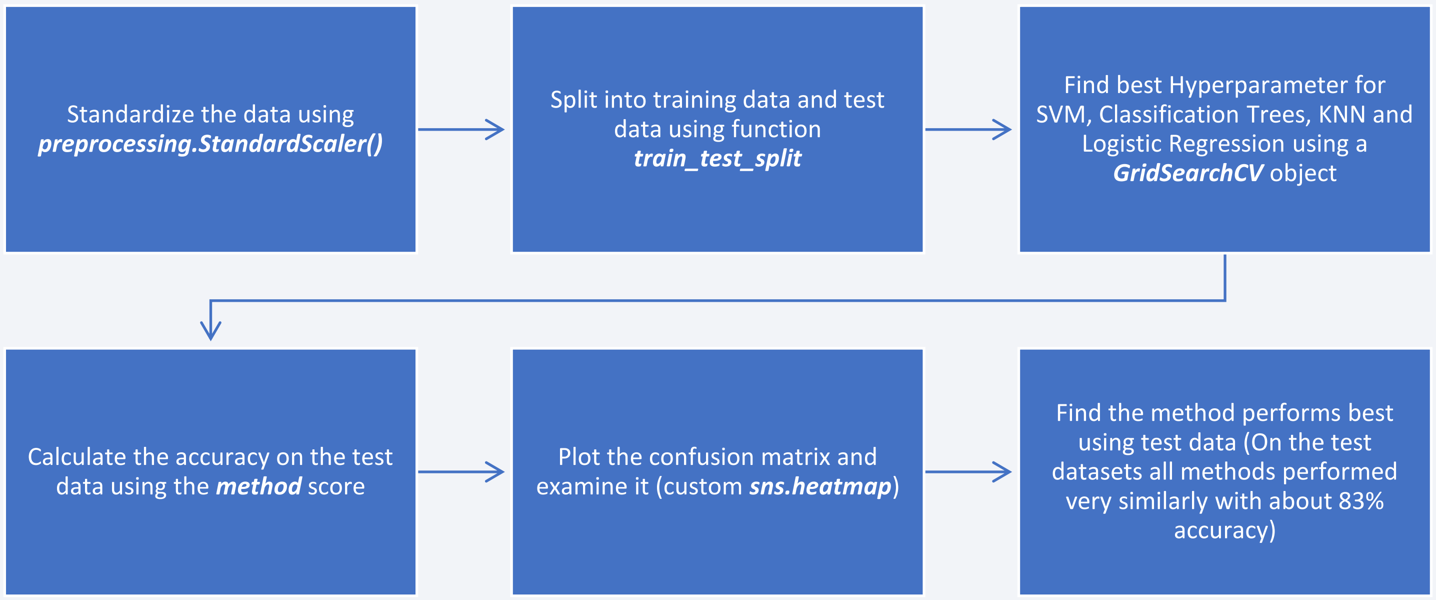
- Which site has the largest successful launches?
- Which site has the highest launch success rate?
- Which payload range(s) has the highest launch success rate?
- Which payload range(s) has the lowest launch success rate?
- Which F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest launch success rate?

We have created an interactive dashboard with a **pie chart** showing successful launches for one or all sites and a **scatter plot** showing successful or unsuccessful launches depending on payload mass. Payload mass range(slider) and launch site(dropdown menu) are made as customizable filters.

GitHub URL of completed Plotly Dash lab:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W3_L2_spacex_dash_app.py

Predictive Analysis (Classification)



GitHub URL of completed predictive analysis lab:

https://github.com/GenBravo/IBM-Course-10-Applied-DS-Capstone-Project/blob/f66d50835d69f9b1ba7a9a8090373ee60fbf17b8/W4_L1_IBM-DS0321EN-SkillsNetwork_labs_module_4_SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

Exploratory data analysis results:

FlightNumber vs. PayloadMass

- as the flight number increases, the first stage is more likely to land successfully. The payload mass is also important; it seems the more massive the payload, the less likely the first stage will return.

FlightNumber vs LaunchSite

- higher flight numbers generally are more successful

Payload Vs. Launch Site

- for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).

FlightNumber and Orbit type

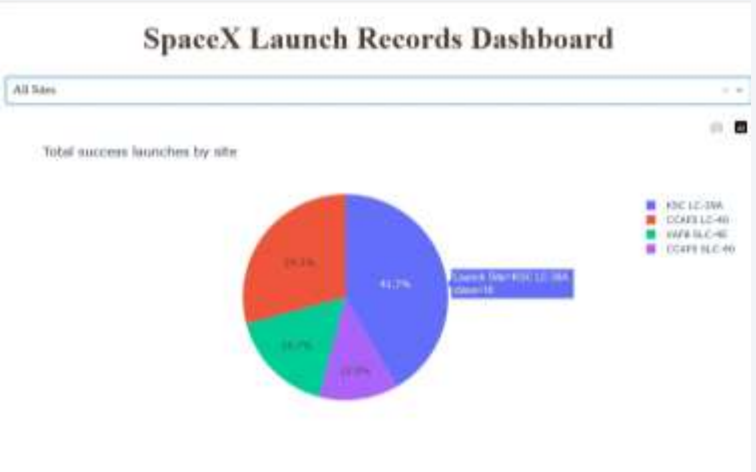
- in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

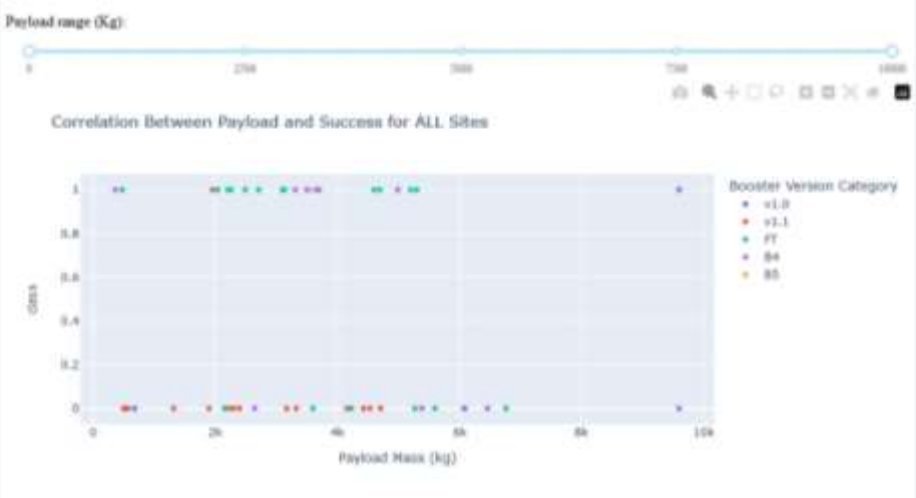
Payload and Orbit type

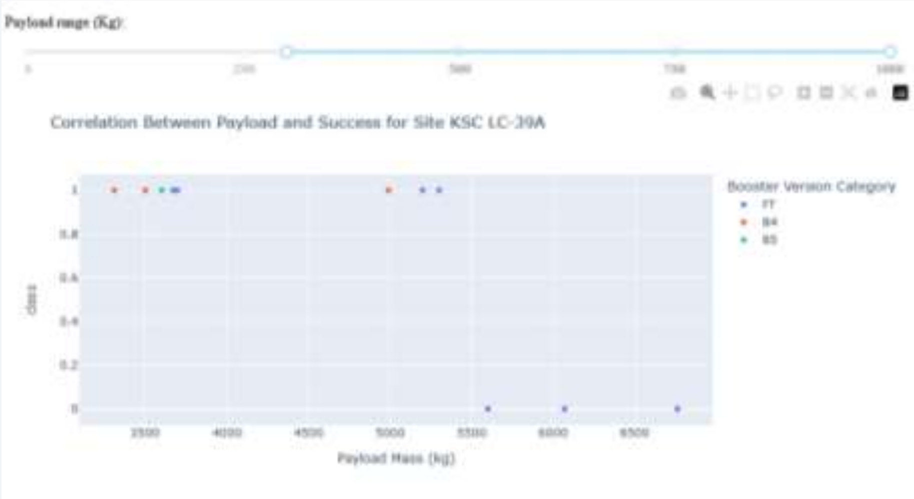
- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Results

Interactive analytics demo in screenshots:







Results

Predictive analysis results

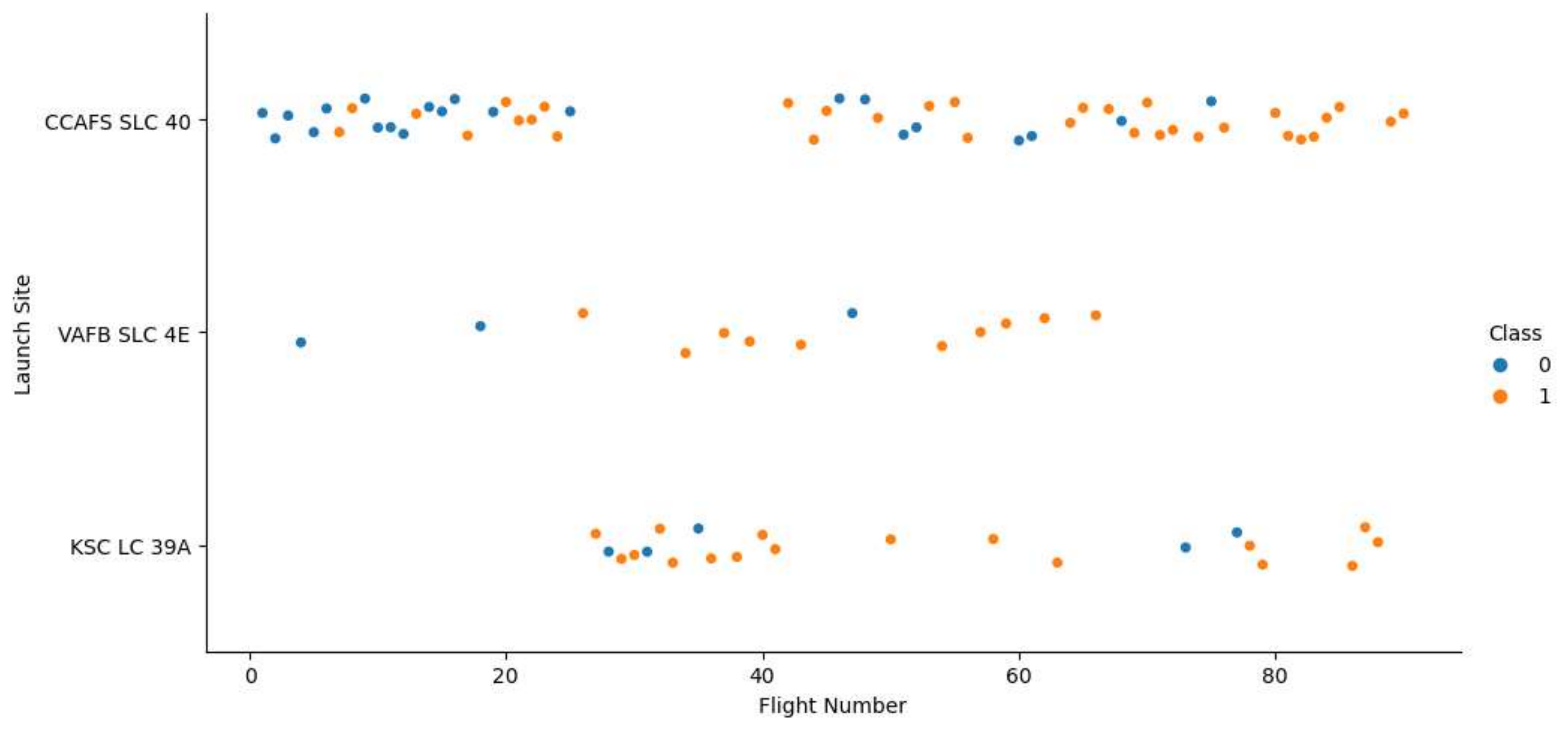
After fitting and finding the best Hyperparameters for SVM, Classification Trees, KNN and Logistic Regression algorithms, they gave a very similar accuracy, confusion matrix, precision, recall, and F1 score results.

In such cases we are free to use any model based on other considerations, such as Training and Prediction Speed

Section 2

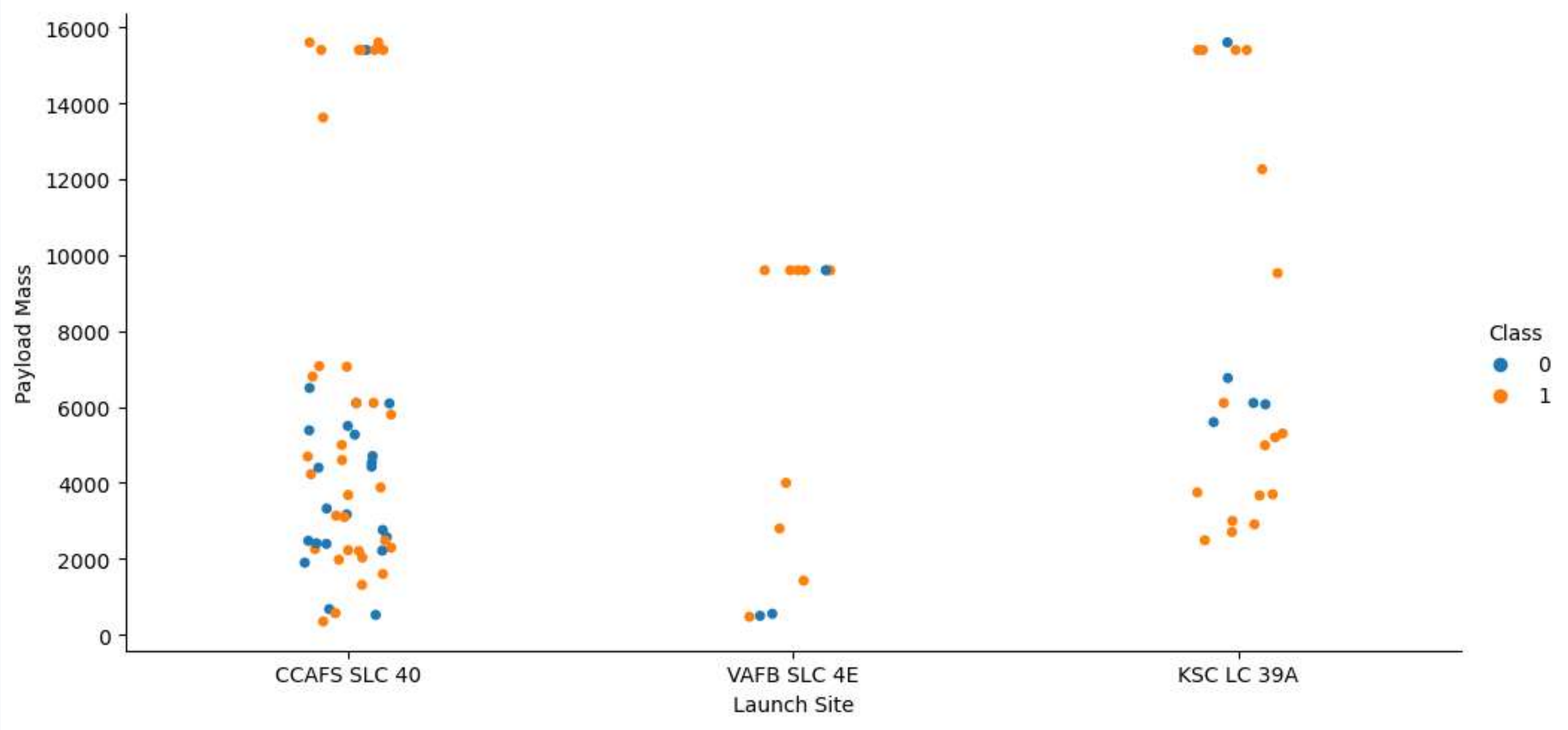
Insights drawn from EDA

Flight Number vs. Launch Site



- Launch sites tend to have more successful outcomes with more flights, except in cases where there were large pauses in using a site

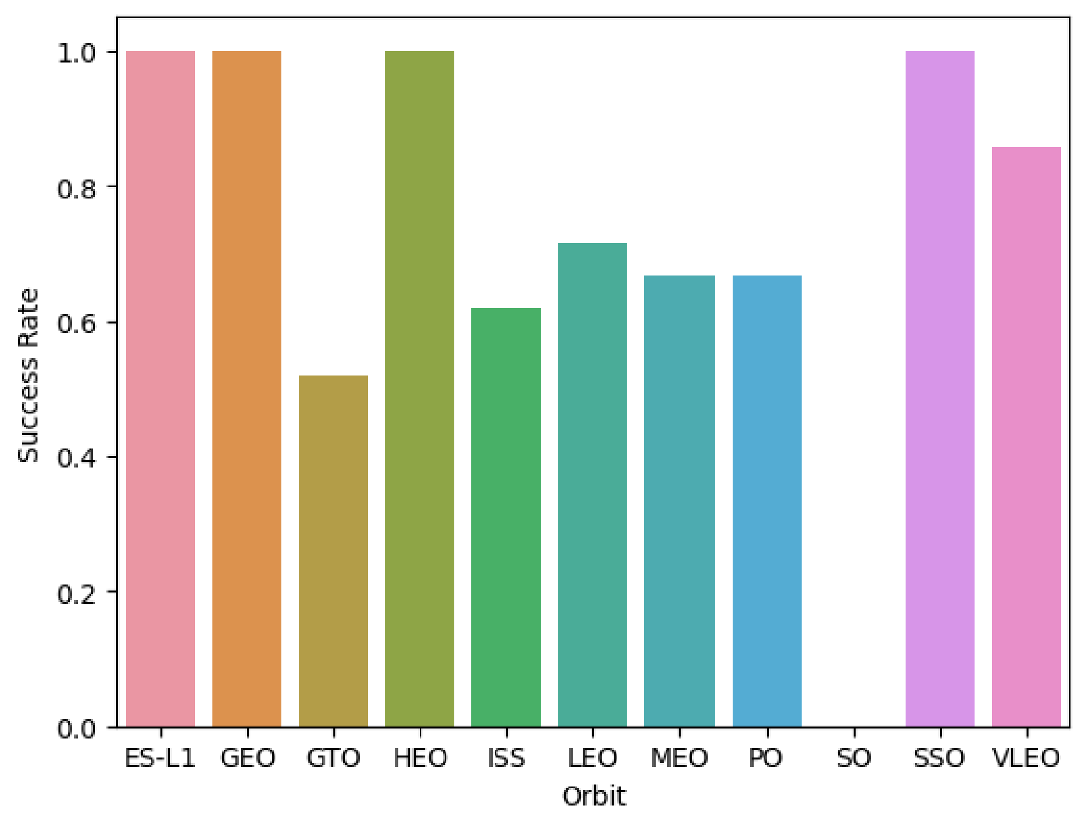
Payload vs. Launch Site



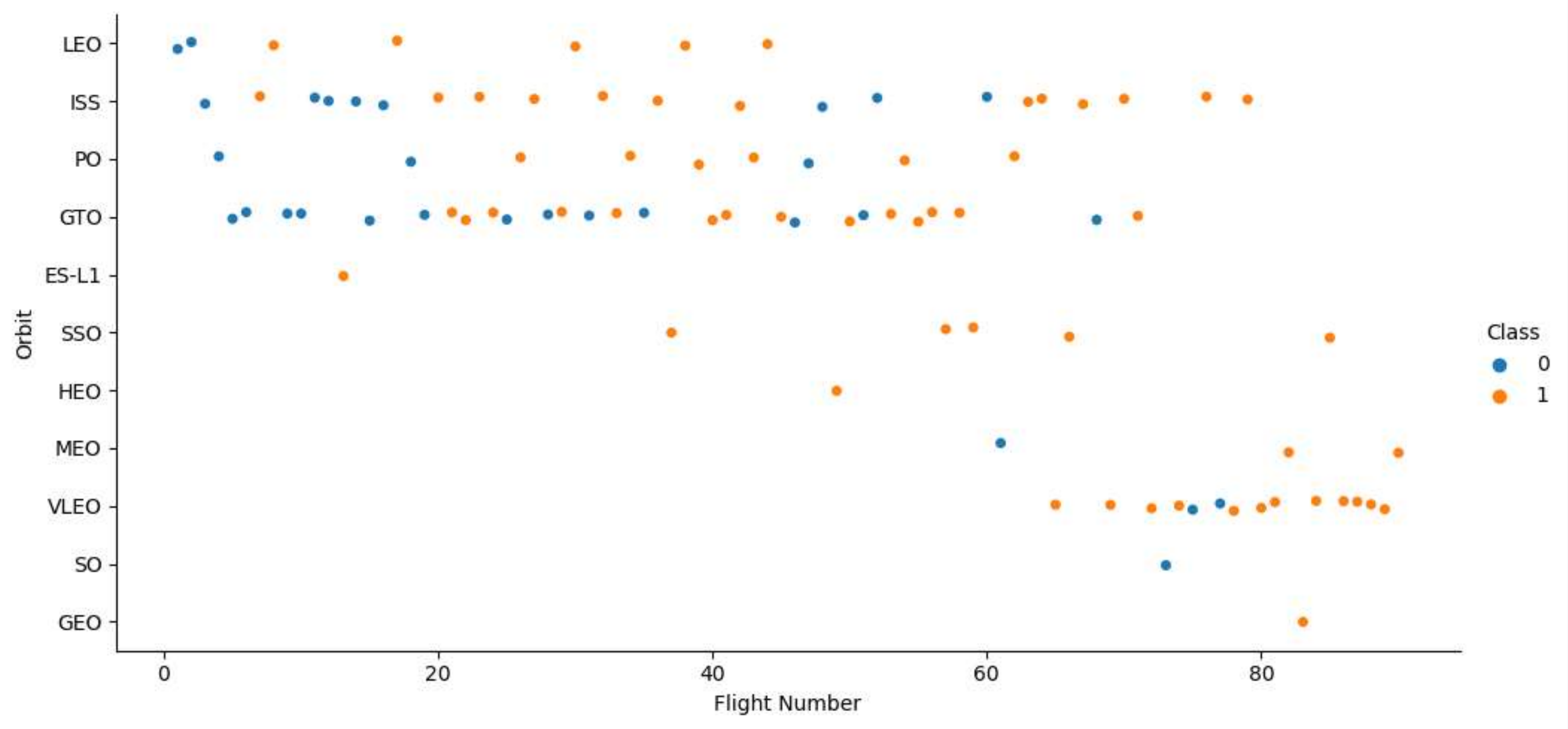
- For the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000)

Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO and VLEO seem to be significantly more likely orbits to have a successful launch

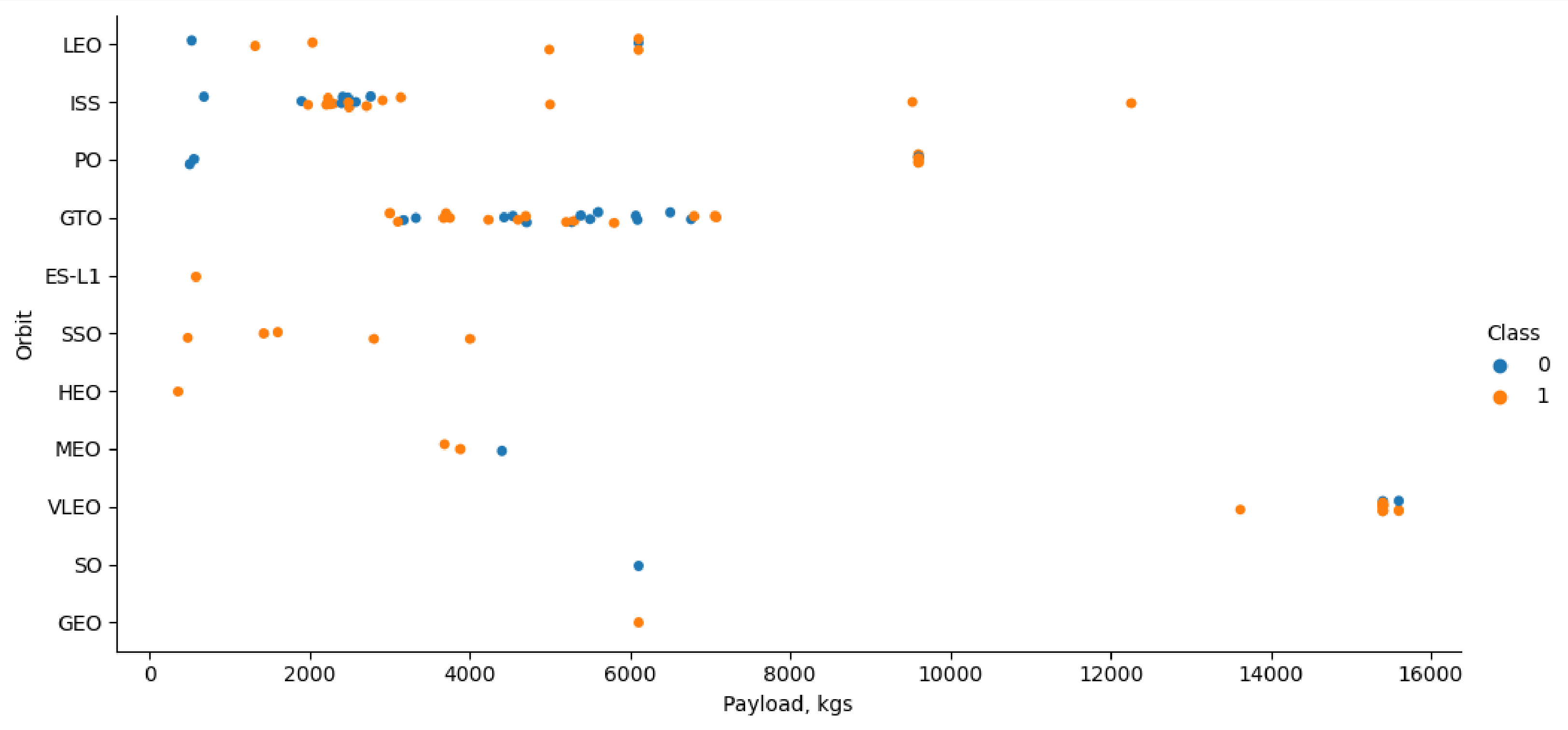


Flight Number vs. Orbit Type



- In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit

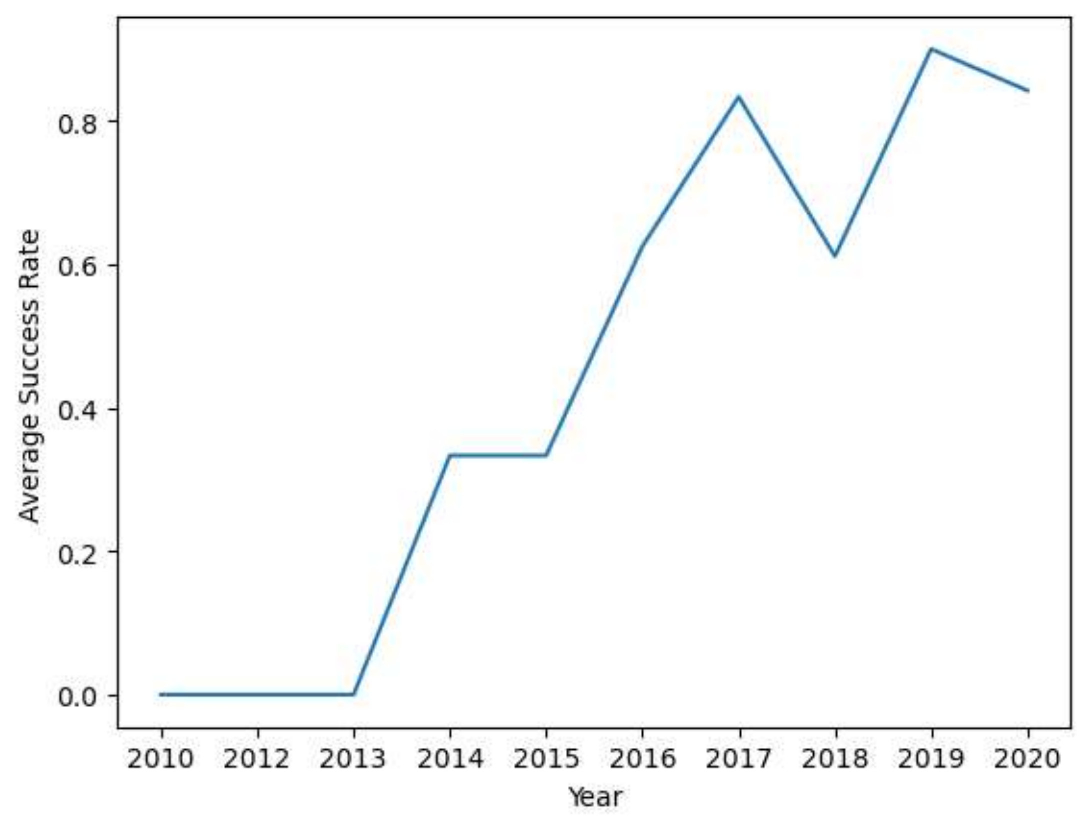
Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

- The sucess rate since 2013 kept (mostly) increasing till 2020



All Launch Site Names

- Find the names of the unique launch sites

```
In [8]: %sql SELECT DISTINCT(Launch_Site) FROM SPACEXTBL

 * sqlite:///my_data1.db
Done.
```

Out[8]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`

In [18]:
```
%%sql
SELECT *
FROM SPACEXTBL
WHERE Launch_Site LIKE ('CCA%')
LIMIT 5
```

```
 * sqlite:///my_data1.db
Done.
```

Out[18]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS_KG_	Orbit	Customer	Mission_Outcome	Landing_O
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (pa
2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (pa
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

Display the total payload mass carried by boosters launched by NASA (CRS)

In [20]:

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)'
```

```
 * sqlite:///my_data1.db
Done.
```

Out[20]:

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

Display average payload mass carried by booster version F9 v1.1

In [23]:

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE Booster_Version = 'F9 v1.1'
```

```
 * sqlite:///my_data1.db
Done.
```

Out[23]:

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

```
In [27]: %%sql
SELECT MIN(Date)
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)'

 * sqlite:///my_data1.db
Done.
```

Out[27]:

MIN(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

In [31]:

```
%%sql
SELECT Booster_Version
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (drone ship)'
AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

```
 * sqlite:///my_data1.db
Done.
```

Out[31]:

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

List the total number of successful and failure mission outcomes

In [36]:
```
%%sql
SELECT Mission_Outcome, COUNT(*)
FROM SPACEXTBL
GROUP BY Mission_Outcome
```

```
 * sqlite:///my_data1.db
Done.
```

Out[36]:

Mission_Outcome	COUNT(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

In [39]:
```
%%sql
SELECT DISTINCT(Booster_Version)
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

```
 * sqlite:///my_data1.db
Done.
```

Out[39]:

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
In [58]:
%%sql
SELECT Date, "Time (UTC)", substr(Date, 6, 2) AS MONTH ,substr(Date,1,4) AS YEAR, Landing_Outcome, Booster_Version, Launch_S
FROM SPACEXTBL
WHERE Landing_Outcome = 'Failure (drone ship)'
AND Date BETWEEN '2015-01-01' AND '2015-12-31'

 * sqlite:///my_data1.db
Done.
```

Out[58]:

Date	Time (UTC)	MONTH	YEAR	Landing_Outcome	Booster_Version	Launch_Site
2015-10-01	09:47:00	10	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	20:10:00	04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

In [70]:
```
%%sql
SELECT Landing_Outcome, COUNT(*) AS 'Count'
FROM SPACEXTBL
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Count DESC
```

```
 * sqlite:///my_data1.db
Done.
```

Out[70]:

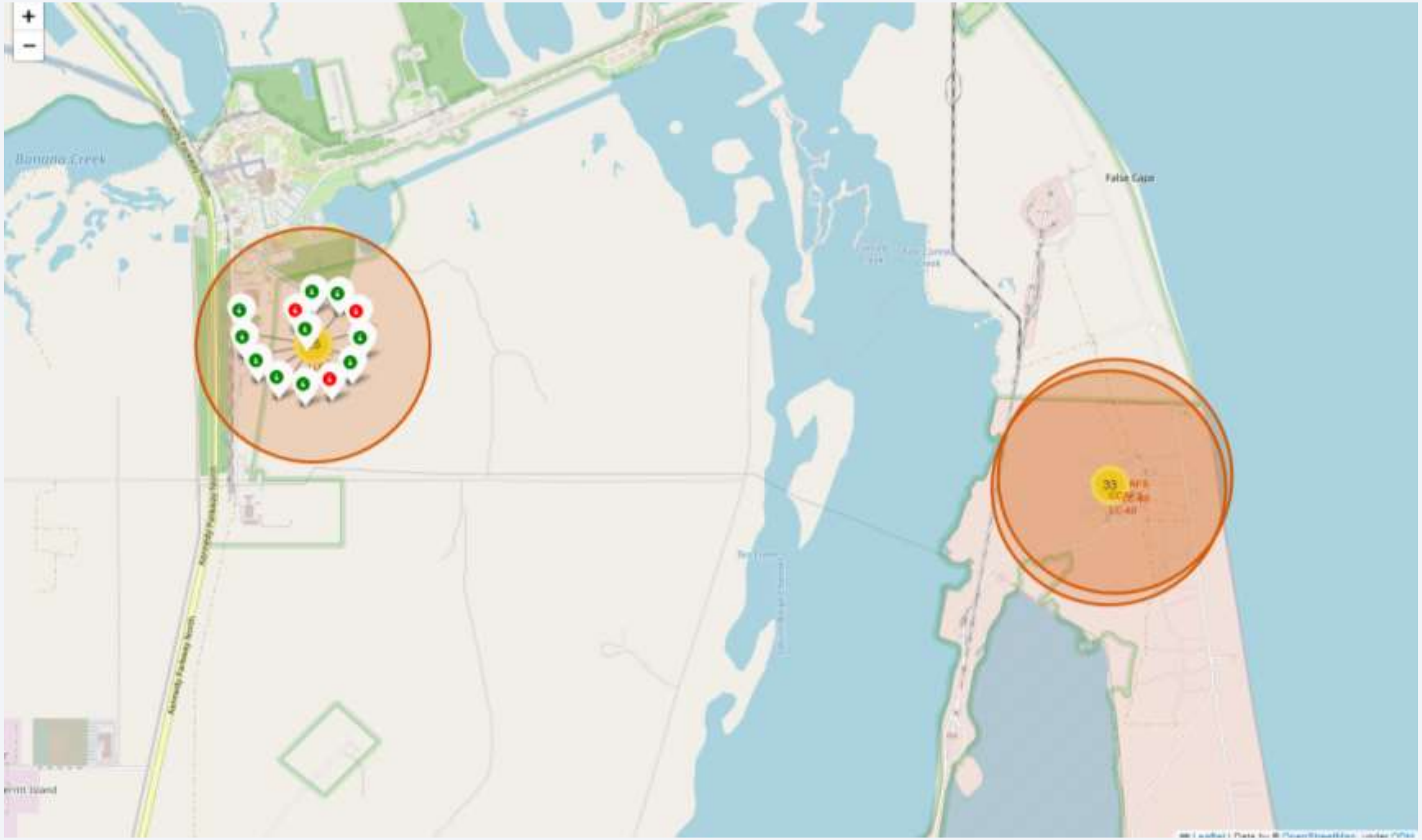
Landing_Outcome	Count
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

Folium. Mark all launch sites on a map

Folium. Mark the success/failed launches for each

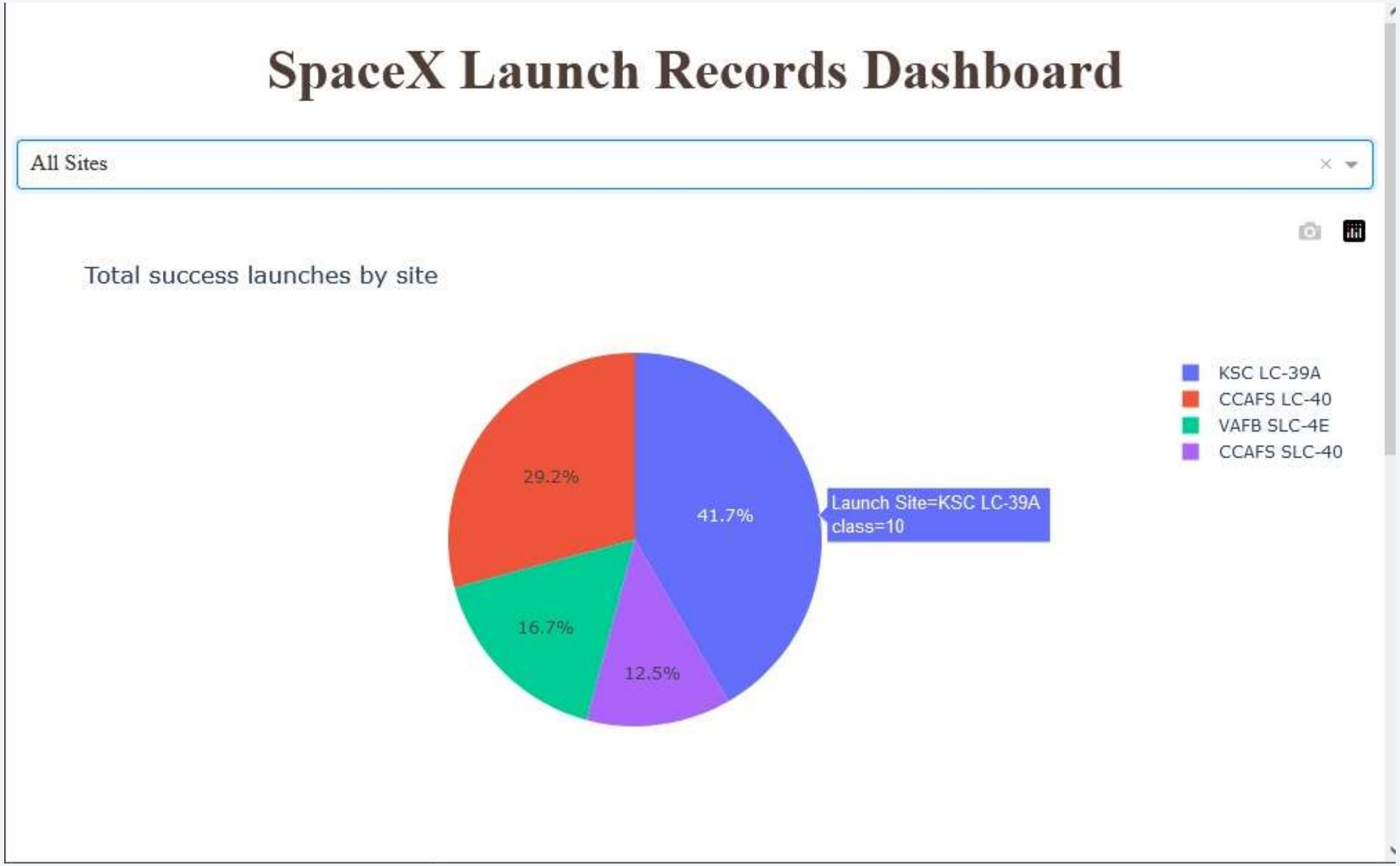


Folium. Calculate the distances between a launch site to its proximities

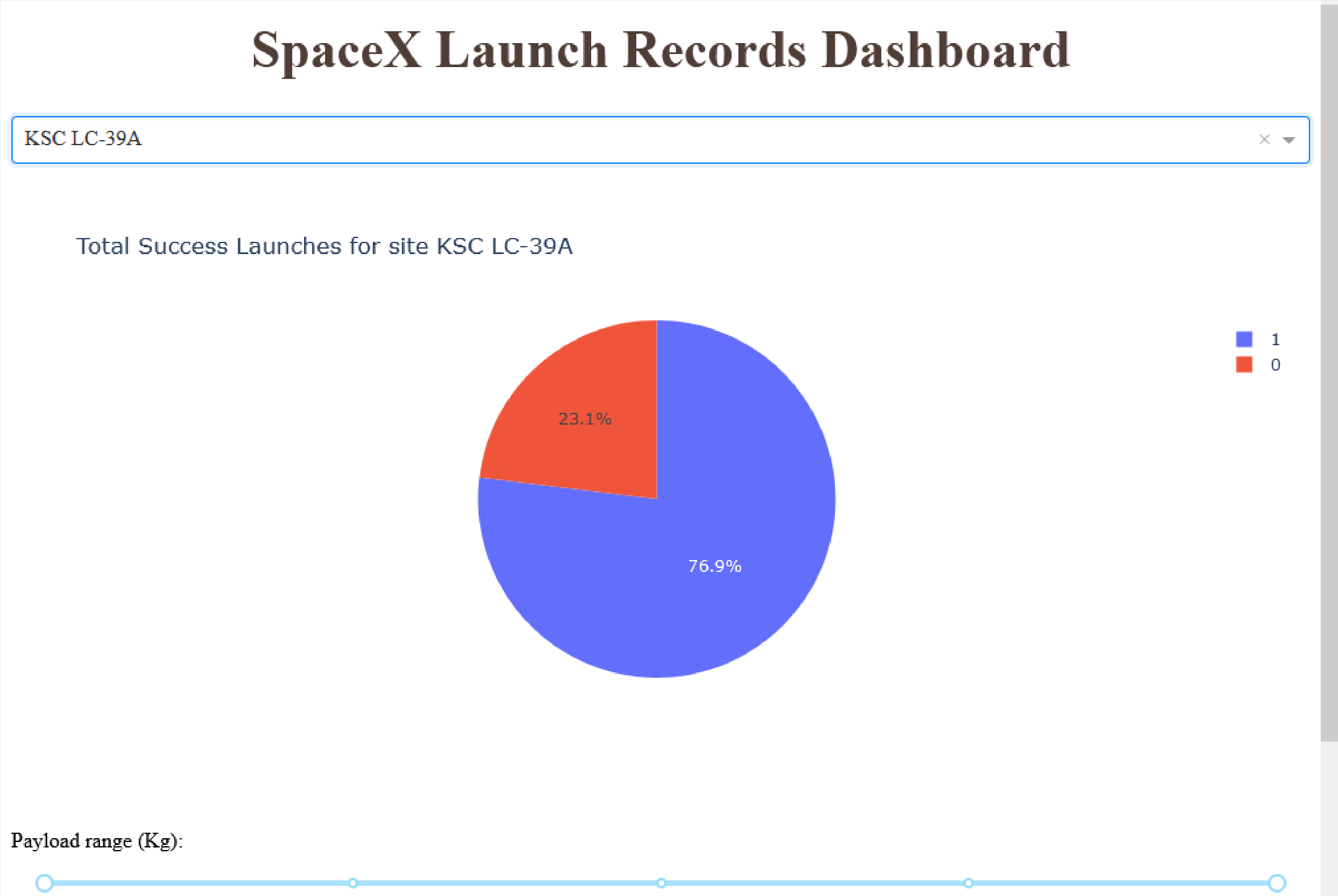
Section 4

Build a Dashboard with Plotly Dash

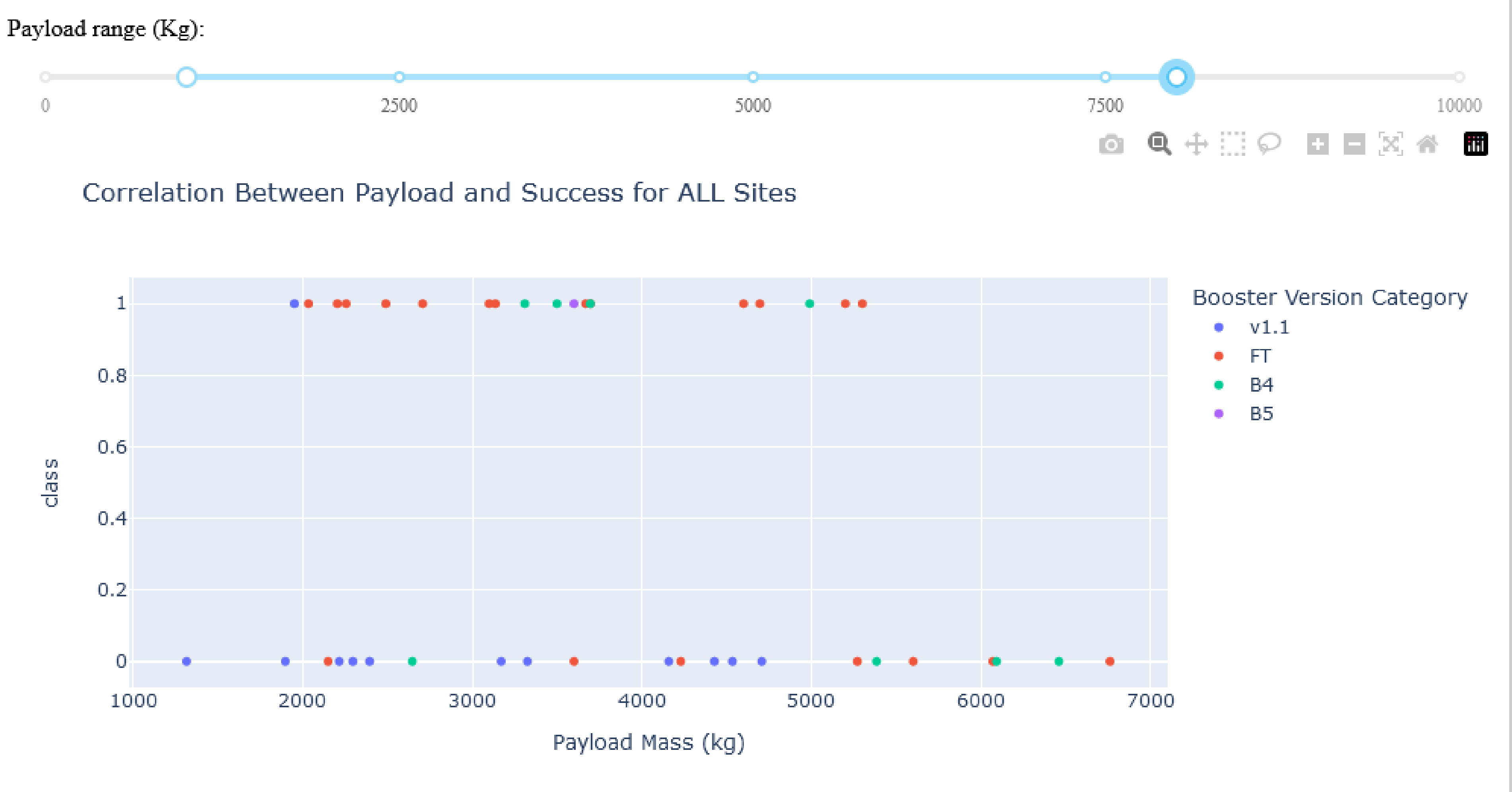
Dashboard. Launch success count for all sites, in a piechart



Dashboard. Piechart for the launch site with highest launch success ratio



Dashboard. Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider



Section 5

Predictive Analysis (Classification)

Classification Accuracy

- All tested models performed with 83% accuracy on the test dataset

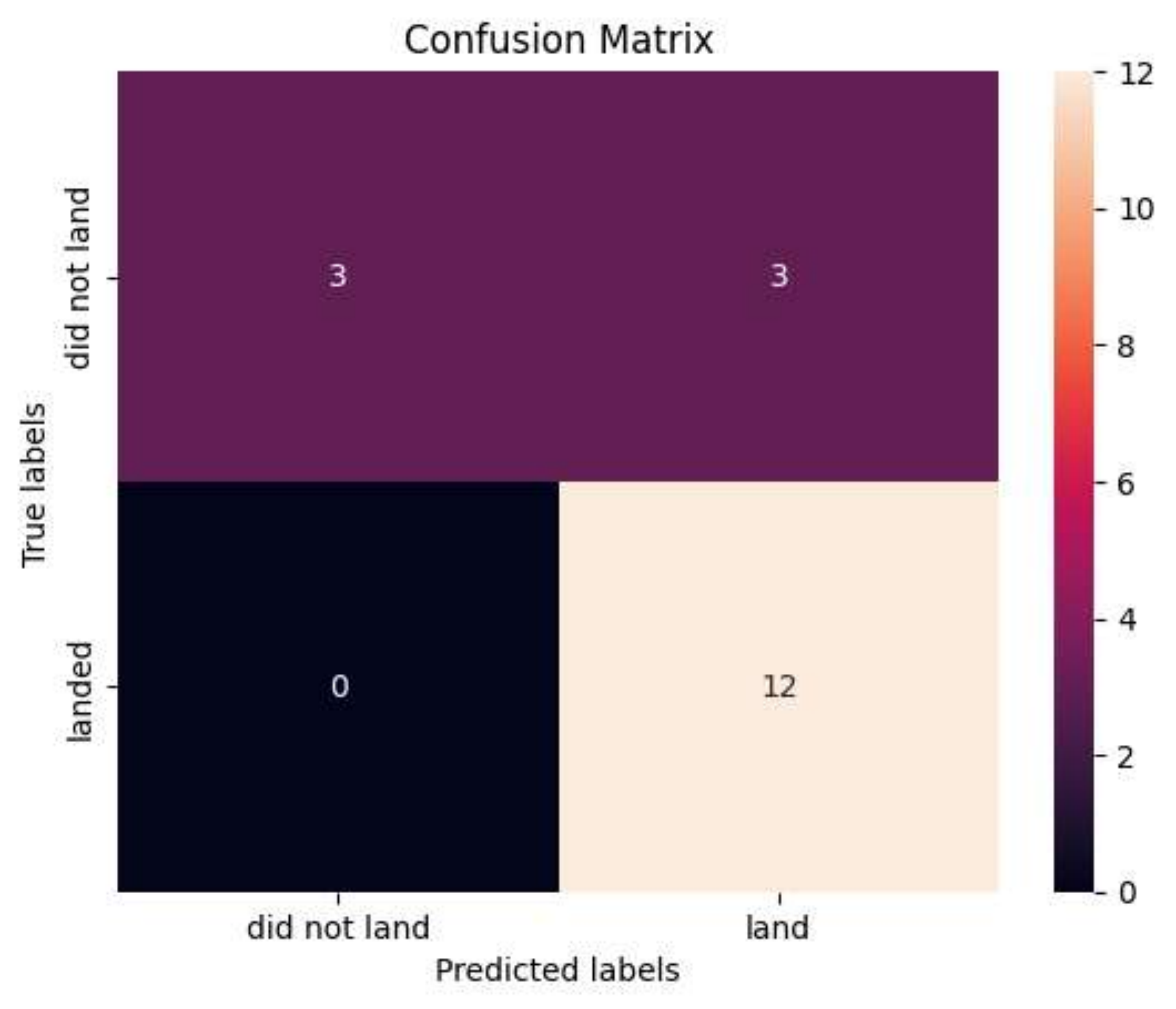
```
svm_cv.score(X_test, Y_test)
0.8333333333333334

logreg_cv.score(X_test, Y_test)
0.8333333333333334

tree_cv.score(X_test, Y_test)
0.8333333333333334

knn_cv.score(X_test, Y_test)
0.8333333333333334
```

Confusion Matrix



All models performed similarly and created the same confusion matrix

Conclusions

- We have started this project with collecting data. Our two sources were SpaceX API and scraping Wikipedia
- We then examined, cleaned and wrangled obtained data with the help of Pandas and Numpy libraries
- We then moved on to exploring the data (EDA) with SQL queries and visualizations using the Seaborn and MatPlotLib libraries to build graphs
- Next we moved on to interactive visualizations by plotting locations on an interactive map using Folium library and building a dashboard using Dash and Plotly
- Finally after engineering useful features we moved on to modeling the data using different classifications algorithms and comparing their effectiveness

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!